ECE 657A Assignment #2

Yuzhou Wang (20609396), Laura McCrackin (20262085),
and Huang Tianhui (20587328)

March 11, 2016

1 Parameter Selection and Classification

1.1 Preprocessing

For the original dataset D, we observed that there are no missing values and that the values for all points are close to the mean, so we simply used Z-score normalization to preprocess the data without any further outlier correction. As directed, we use the first half of data for training, and the second half for testing.

1.2 K-NN

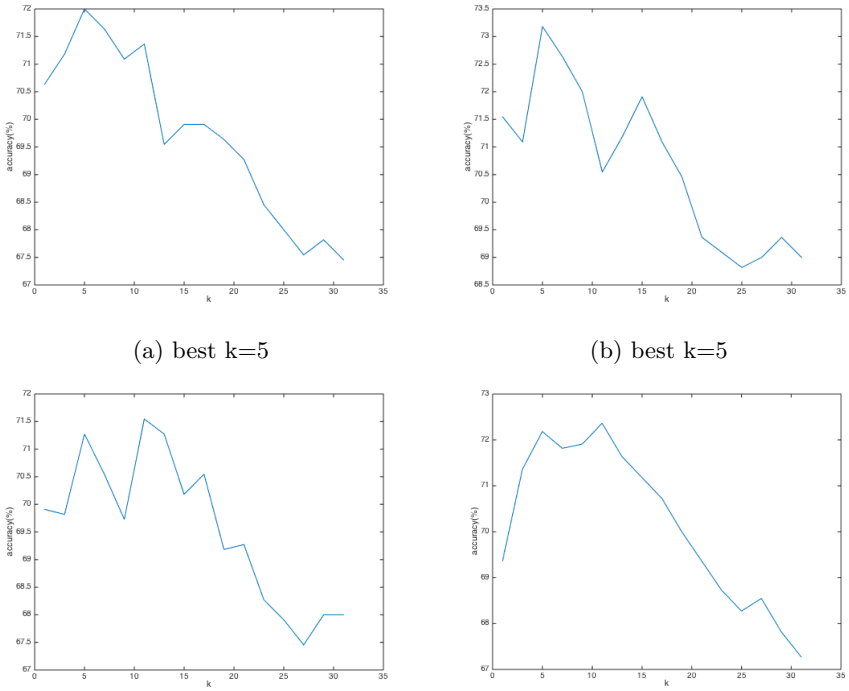
To perform 5-fold cross validation, the training data was divided randomly into 5 partitions using the "crossvalind" function. Four of these partitions were used for training, with the fifth being using for testing; k-NN was run five times in this manner, so that each of the five training partitions in turn was used as the testing set while the remaining partitions were used as training. The cross-validation accuracy was then taken to be the average accuracy for each of these five trials.

This cross validation process was performed for each k value, k=1,3,5...31. It should be noted that since the cross validation partitioning is random, the k value producing the highest classification accuracy is not the same every time this experiment is performed.

In Figure 1, the classification accuracy for k-NN can be observed for four sample runs. As can be seen, each time the value of best k changes; it is consistently between 5-12 with an accuracy of between 71% and 73%.

1.3 SVM with RBF Kernel

Next, we perform a grid search to determine the best combination of c and *gamma* values for classifying our dataset using an SVM with a radial basis function (RBF) kernel. That is, we perform 5-fold cross-validation for each combination of candidate c and *gamma* values, for c = 0.1, 0.5, 1, 2, 5, 10, 20, 50 and *gamma* = 0.01, 0.05, 0.1, 0.5, 1, 2, 5, 10.



(a) best k=5

(b) best k=5

(c) best k=12

(d) best k=12

Figure 1: Different cross-validation experiments and their results

The resulting ROC curve may be seen in Figure 2. The optimal combination of parameters can be seen as the innermost point of the curve.

1.4 Comparison of Methods

We then perform a comparison of k-NN, SVM using an RBF kernel, a Naive Bayes classifier, a Decision Tree (using the "fitctree" function), and a Neural Network (using "patternnet"). The SVM and k-NN were run using the aforementioned ideal values, while the remaining methods were used with default parameters. All experiments were repeated 20 times, with the classification accuracy averaged for all trials.

From the accuracy table above, we can see that the Decision Tree provides the highest accuracy, precision, recall and F-measure values, but also requires the most time to run. The worst performance is achieved using k-NN.

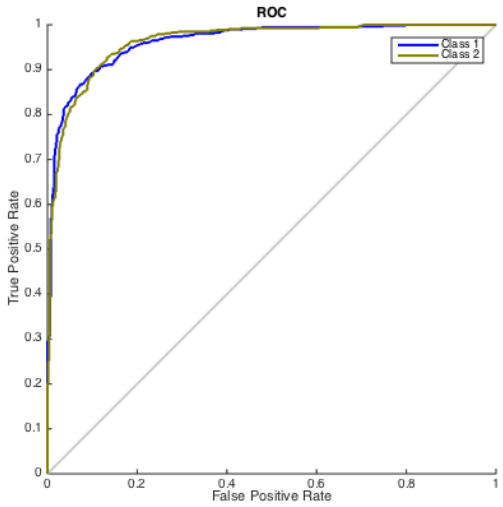


Figure 2: ROC

	K-NN	RBF SVM	Naive Bayes	Decision Tree	Neural Net
Accuracy	74.4182	89.1864	86.4864	92.9818	82.5773
std accuracy	1.2322	0.7382	0.6734	0.9843	1.7322
Precision	0.9104	0.9113	0.8603	0.9391	0.8461
std Precision	0.028	0.0131	0.0101	0.0146	0.0240
Recall	0.5415	0.8764	0.8810	0.9250	0.8089
std recall	0.0293	0.0097	0.0104	0.0126	0.0330
F-measure	0.6782	0.8934	0.8705	0.9319	0.8265
std F-measure	0.0195	0.0070	0.0060	0.0093	0.0186
Training time (s/sample)	0.0081	0.1028	0.0093	0.00518	0.2698
Class. time (s/sample)	0.008	0.0690	0.0026	0.0104	0.0082

1.5 Strengths and Weaknesses

For test accuracy, it could be reflected by accuracy, precision and f-measure. At the same time we could also combine the std to see the stable result.Precision is how useful the search results are, and recall is how complete the results are.high precision means that an algorithm returned substantially more relevant results than irrelevant, while high recall means that an algorithm returned most of the relevant results.Recall is defined as the number of relevant documents retrieved by a search divided by the total number of existing relevant documents, while precision is defined as the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search. For f-measure, it considers both the precision p and the recall r of the test to compute the score: p is the number of correct positive results divided by the number of

all positive results, and r is the number of correct positive results divided by the number of positive results that should have been returned. The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst at 0.

From the above table, we could get the conclusion that Decision tree method could get the highest accuracy. The std accuracy is also has a relatively low level, so it means the average result is relatively stable. For the precision and std precision, all of the methods have a good performance, but maybe the best one is still the decision tree method.For recall and std recall, we could also get the conclusion that decision tree method and neural network have better performance, which means that the true positive divided by the correct answer for both true positive and false negative is high. Similarly, the decision tree method also have the good performance for f-measure and std-measure.

For classification time in K-NN, it should compare all test samples with test samples. For time percepective, naive bayes has the best performance and also have the lowest time value.

So we could choose decision tree method for both time saving and higher accuracy.

2 Clustering Analysis

2.1 Hierarchical Clustering Using Agglomerative Algorithms

2.1.1 Comparison of Linkage Methods

We use the Matlab library function 'clusterdata' to perform clustering using single, complete, and ward linkage methods, respectively.

To find the separation index, we calculate the centre of each cluster and calculate the square of the distance from each point to the centre point. This result is then divided by the total number of rows multiplied by the maximum distance for each cluster. For rand-index, it should be calculated by $(a+d)/M$, where a is the number of True Positives and d is the number of False Negatives, with M being the total number of possible point comparisons. For F-measure, we first calculate the precision and recall, and then use the function: $F(i,j) = 2 * precision(i,j) * recall(i,j)/(precision(i,j) + recall(i,j))$.

According to the definition, the separation index is highly dependant on the clustering results. Smaller separation index means that every point in the cluster is close and as a result there could be more groups. The result of rand-index and f-measure could have the similar trend for representing accuracy of the same cluster group, which is corresponding different with separation-index. It means that if there is a higher accuracy for cluster, the rand-index and f-measure could correspondingly higher, but it means larger size of cluster, as a result separation-index could be lower.

There are two ways to calculate the separation index; one is for the maximum distance and another is for minimum distance. Below the first table shows the result for the maximum distance. For maximum distance method:

	Single	Complete	Ward
Separation-Index	1.7540	0.6151	0.4937
Rand-Index	0.1097	0.8184	0.8772
F-measure	0.0962	0.2168	0.2801

When we choose the maximum method, the smaller the value it is, the closer for the points are within a class and the greater the separation from other classes. A smaller SI value is therefore ideal. The opposite is true for rand-index and f-measure. So, the ward algorithm has the best result, having the smallest separation-index and the largest rand-index and f-measure values.

2.1.2 Optimal Number of Clusters

After we run the 'ward' algorithm for a variety of cluster numbers, we found the optimal value to be 15.

2.2 Clustering Using K-Means

The result of using k-means can be seen in Figure 4a. We could see that both rand-index increase a lot with the increasing number of cluster, and f-measure increase a lot at the beginning and later a little bit decrease, but separation-index decrease.

The reason why rand-index increase is because of the calculating function with the increasing number of cluster, it is closer to the right cluster way, but for the value of M it did not change, so it has an increasing trend. Similarly, f-measure also increase, for increasing number of clusters, it could reflect better accuracy. And for the separation-index, with the increasing number of cluster the value of SI should also decrease for more densy in the same group and more distance for different class. Above all, all the three methods shows that the increasing number of clusters leading to a better result.

Combined with three methods, the best cluster number is 8, without too high separation-index or too low rand-index and f-measure.

2.3 Fuzzy C-Means

2.3.1 Cluster Membership

For the fuzzy c-means methods, we could use 'fcm' function directly, and label the result is same with digit '1' or '3', so that we could get the graphs in Figure 5.

From the graph, we could see that, for digit 1, it mainly in cluster 5 to 7, without too much overlapping with other groups. However, the result of digit 3 could be checked in different clusters, which means that there are more overlaps in cluster 3 but less in cluster 1. For the reason why there is less overlapping for digit 1 maybe for the hand writing dataset, the digit '1' is hard to make confusing with other digits unlike digit '3', which is similar to the digit 8, for example.

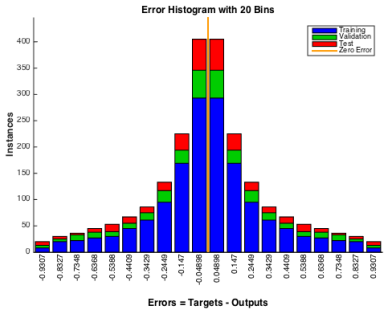
2.3.2 Hard Clustering

In fuzzy clustering (also referred to as soft clustering), data elements can belong to more than one cluster, and associated with each element is a set of membership levels. But in hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. Hard clustering can perhaps be considered a less flexible method, where the subtleties of classification uncertainty reflected in fuzzy clustering are no longer retained.

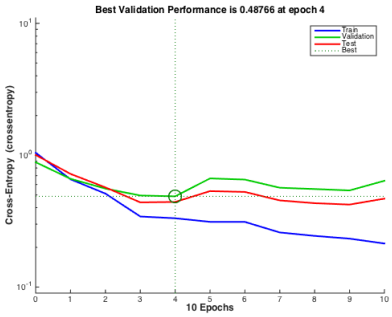
For Fuzzy c-means
Separation-index=0.4454
Rand-Index=0.8905
F-measure=0.2844

For linkage method=single
Separation-Index=1.7540
Rand-Index=0.1097
F-measure= 0.0962
For linkage method=complete
Separation-Index=0.6151
Rand-Index=0.8184
F-measure=0.2168
For linkage method=ward
Separation-Index=0.4937
Rand-Index=0.8772
F-measure= 0.2801

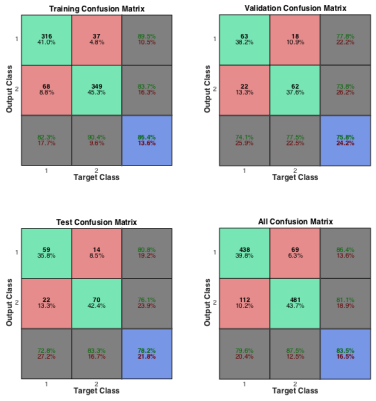
As can be seen above, fuzzy-c means has better performance than the other three methods.



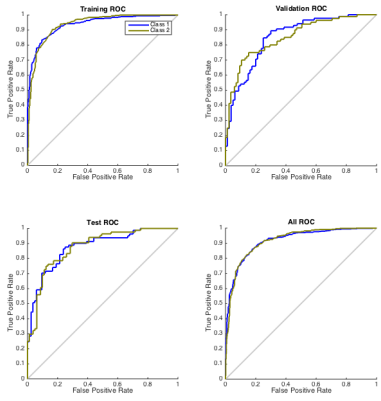
(a) netural-error



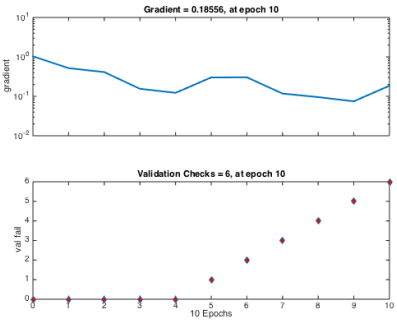
(b) perfomance



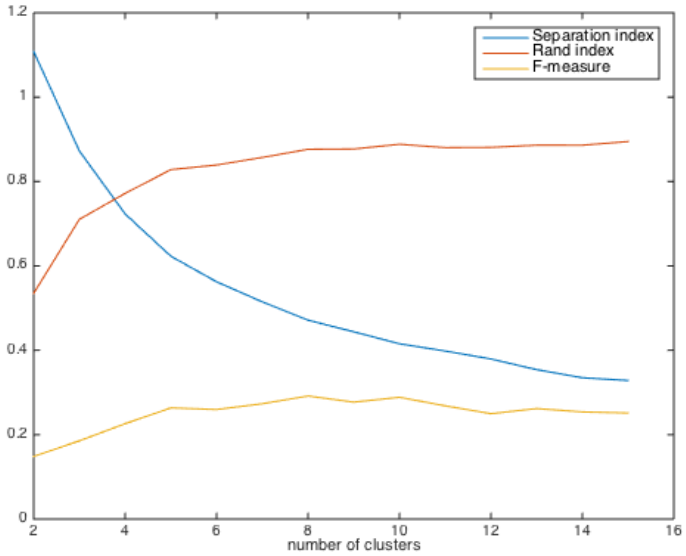
(c) neural-confusion



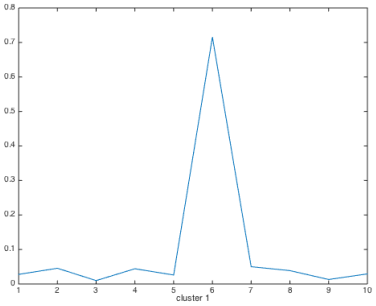
(d) neural-receiver



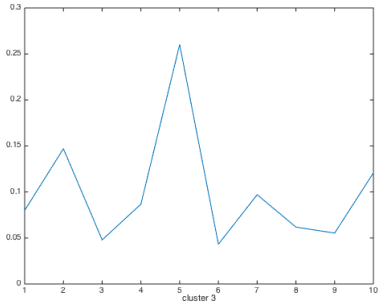
(e) neural-gradient



(a) K-means algorithm



(a) Digit 1



(b) Digit 3

Figure 5: Points from digits 1 and 3 in each cluster (out of 1.0)